UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/822,454	04/12/2004	Erez Haba	MSFTP641US	2824

27195 7590 06/02/2008

AMIN. TUROCY & CALVIN, LLP
24TH FLOOR, NATIONAL CITY CENTER
1900 EAST NINTH STREET
CLEVELAND, OH 44114

EXAMINER	
WANG, BEN C	

ART UNIT	PAPER NUMBER
2192	

NOTIFICATION DATE	DELIVERY MODE
06/02/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

docket1@thepatentattorneys.com
hholmes@thepatentattorneys.com
lpasterchek@thepatentattorneys.com

Office Action Summary	Application No.	Applicant(s)	
	10/822,454	HABA ET AL.	
	Examiner	Art Unit	
	BEN C. WANG	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 April 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-5,7-12 and 14-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5,7-12 and 14-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on April 15, 2008 has been entered.

2. Applicant's amendment dated April 15, 2008, responding to the Office action mailed November 1, 2007 provided in the rejection of claims 1-27, wherein claims 1, 10, 14, 17, and 23 are amended, and claims 6 and 13 have been canceled.

Claims 1-5, 7-12, and 14-27 remain pending in the application and which have been fully considered by the examiner.

Applicant's arguments with respect to claims rejection have been fully considered but are moot in view of the new grounds of rejection – see Murakami and Cowan - arts made of record, as applied hereto.

Claim Rejections – 35 USC § 102(a)

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102(a) that form the basis for the rejections under this section made in this office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

3. Claims 17 and 22 are rejected under 35 U.S.C. 102(a) as being anticipated by Murakami et al., (Pub. No. US 2003/0167423 A1) (hereinafter 'Murakami' - art made of record)

4. **As to claim 17** (Currently Amended), Murakami discloses a test management methodology comprising:

- retrieving metadata regarding test version information in relation to software code version under test (e.g., Fig. 4, elements 150 – Library Consistency Test Unit, 152 – Attribute Extraction Unit; 153 – Consistency Test Unit; [0049] - the library consistency test unit tests the consistency among library resources, including: source files, object files, and executive files …; Fig. 7, items 2 – "source", 3 – "object"; Abstract, Lines 6-9 – The record includes file location of the source file, and information … will be updated each time the source file is modified; [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …)
- persisting the metadata to a markup language file versioned with test assets and source code (e.g., Fig. 14, elements 82 – Source File Attribute Record (XML), element 71 – Object File Attribute Record (XML); Fig. 5 – 121c – Source File Attribute Record Filed; Fig. 6, 131c – Object File Attribute Record Field);

- continuously modifying test information such that new features are added and/or removed to test version changes to the software code under test (e.g., [0049], Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management); and
- generating test results that are tagged with test version information in relation to software code version under test, the test results and test version information are all version tagged data and dependent on the versions of the software code under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …; [0049], Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management; [0007], Lines 5-11 – those library resources, however, are modified for various reasons … implementation of additional capabilities … the modified source file is then recompiled, and the resulting new object file is used to create a new version of executive files).

5. **As to claim 22** (original) (incorporating the rejection in claim 17), please refer to claim **17** as set forth above accordingly.

Claim Rejections – 35 USC § 103(a)

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-5, 9, 14, and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Murakami in view of Kenneth W. Cowan (Pat. No. US 7,243,337 B1) (hereinafter 'Cowan' - art made of record)

7. **As to claim 1** (Currently Amended), Murakami discloses an application test management system comprising:

- a version component that monitors source under test components (Fig. 14, element 82 – Source File Attribute Record (XML)) and test components (Fig. 14, element 71 – Object File Attribute Record (XML) Fig. 5 – 121c – Source File Attribute Record Filed; Fig. 6, 131c – Object File Attribute Record Field); for changes (e.g., Fig. 4, elements 150 – Library Consistency Test Unit, 152 – Attribute Extraction Unit; 153 – Consistency Test Unit; [0049] - the library consistency test unit tests the consistency among library resources, including: source files, object files, and executive files …; Fig. 7, items 2 – "source", 3 – "object"; Abstract, Lines 6-9 – The record includes file location of the source file, and information … will be updated each time the source file is modified; [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …);
- a build drop component that comprises an executable version of the software under test and includes changed data from the version component (e.g., [0049],

Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management);

- a test case file component that includes metadata associated with test components and source under test components received from the version component that indicates relationships between versions of source under test components and versions of test cases, the test case file component includes attributes necessary for query and test management and is continuously modified such that new features are added and/or removed to test changes in the source under test components, source under test components represent specific versions of source code (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …; [0049], Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management; [0007], Lines 5-11 – those library resources, however, are modified for various reasons … implementation of additional capabilities … the modified source file is then recompiled, and the resulting new object file is used to create a new version of executive files);

Murakami disclose the association between a new executive file version and its corresponding source file version (e.g., [0008]), but does not explicitly disclose the followings:

- a test catalog that provides a repository for a collection of test case files, test cases, and test variations;

- wherein the test case file component generates test results that are tagged with the versions of the source under test components and saved to a data store for historical analysis, the test results and version component are all version tagged data and dependent on the versions of the software under test; and
- wherein the test case file component is loaded into memory or treated as a database to facilitate management operations including at least one of query, reporting, suite composition and scheduling.

However, in an analogous art of *Managing Hardware and Software Configuration Information of Systems Being Tested*, Cowan discloses the followings:

- a test catalog that provides a repository for a collection of test case files, test cases, and test variations (e.g., Col. 7, Lines 35-42 - … enable one to identify a particular version of the software component or library … a version number … other information which may be used to distinguish one version of the software component from another);
- wherein the test case file component generates test results that are tagged with the versions of the source under test components (e.g., Fig. 10, elements 96 – gather software module information, 98 – gather system configuration information) and saved to a data store for historical analysis, the test results and version component are all version tagged data and dependent on the versions of the software under test (e.g., Fig. 10, element 100 – create and store corresponding database records); and

- wherein the test case file component is loaded into memory or treated as a database to facilitate management operations including at least one of query, reporting, suite composition and scheduling (e.g., Fig. 4; Col. 8, Lines 6-9 - … the viewer/query tool accesses platform information and , via a user interface, displays the information in a user-readable form).

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cowan into the Murakami's system to further provide the followings in the Murakami system:

- a test catalog that provides a repository for a collection of test case files, test cases, and test variations;
- wherein the test case file component generates test results that are tagged with the versions of the source under test components and saved to a data store for historical analysis, the test results and version component are all version tagged data and dependent on the versions of the software under test; and
- wherein the test case file component is loaded into memory or treated as a database to facilitate management operations including at least one of query, reporting, suite composition and scheduling.

The motivation is that it would further enhance the Murakami's system by taking, advancing and/or incorporating Cowan's system which offers significant advantages for automatically tracking the various hardware and software components of a particular platform in accordance with a particular version of a software system being tested as once suggested by Cowan (e.g., Col. 3, Lines 26-29)

8. **As to claim 2** (original) (incorporating the rejection in claim 1), Murakami discloses the system wherein the test case file component includes a pointer to the source under test (e.g., Fig. 14, element 82 – Source File Attribute Record (XML))

9. **As to claim 3** (original) (incorporating the rejection in claim 1), Cowan discloses the system wherein the test case file component includes a pointer to requirement for test data (e.g., Fig. 10, elements 96 – gather software module information, 98 – gather system configuration information)

10. **As to claim 4** (original) (incorporating the rejection in claim 1), Cowan discloses the system wherein the test case file component includes a pointer to requirement and/or configuration under test data (e.g., Fig. 10, elements 96 – gather software module information, 98 – gather system configuration information)

11. **As to claim 5** (original) (incorporating the rejection in claim 1), Cowan discloses the system wherein the test case file component includes a pointer to a test case component (e.g., Fig. 10, elements 96 – gather software module information, 98 – gather system configuration information)

12. **As to claim 9** (original) (incorporating the rejection in claim 1), Murakami discloses the system wherein the test case file component is located in the source file under test (Fig. 14, element 82 – Source File Attribute Record (XML))

13. **As to claim 14** (Currently Amended), Murakami discloses a test management system comprising:

- a means for maintaining fine-grained track of a test's relation to a version of software under test (e.g., Fig. 4, elements 150 – Library Consistency Test Unit, 152 – Attribute Extraction Unit; 153 – Consistency Test Unit; [0049] - the library consistency test unit tests the consistency among library resources, including: source files, object files, and executive files …; Fig. 7, items 2 – "source", 3 – "object"; Abstract, Lines 6-9 – The record includes file location of the source file, and information … will be updated each time the source file is modified; [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …);
- a means for continuously modifying test data such that new features are added and/or removed to test version changes to the software under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …; [0049], Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management; [0007], Lines 5-11 – those library resources, however, are modified for various reasons … implementation of additional capabilities … the modified source

file is then recompiled, and the resulting new object file is used to create a new version of executive files); and

- a means for generating test results that are tagged with test version data in relation to the version of software under test, the test results and test version data are all version tagged data and dependent on the versions of the software under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …)

Murakami disclose the association between a new executive file version and its corresponding source file version (e.g., [0008]), but does not explicitly disclose the followings:

- a means for querying test data to facilitate generation of test management reports.

However, in an analogous art of *Managing Hardware and Software Configuration Information of Systems Being Tested*, Cowan discloses the followings:

- a means for querying test data to facilitate generation of test management reports (e.g., Fig. 4; Col. 8, Lines 6-9 - … the viewer/query tool accesses platform information and , via a user interface, displays the information in a user-readable form)

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Cowan into the Murakami's system to further provide the followings in the Murakami system:

- a means for querying test data to facilitate generation of test management reports

The motivation is that it would further enhance the Murakami's system by taking, advancing and/or incorporating Cowan's system which offers significant advantages for automatically tracking the various hardware and software components of a particular platform in accordance with a particular version of a software system being tested as once suggested by Cowan (e.g., Col. 3, Lines 26-29)

14. **As to claim 18** (Previously Amended) (incorporating the rejection in claim 17), Murakami discloses the method wherein version information is retrieved from a version component that monitors changes to source code versions and test versions (e.g., Fig. 4, elements 150 – Library Consistency Test Unit, 152 – Attribute Extraction Unit; 153 – Consistency Test Unit; [0049] - the library consistency test unit tests the consistency among library resources, including: source files, object files, and executive files …; Fig. 7, items 2 – "source", 3 – "object"; Abstract, Lines 6-9 – The record includes file location of the source file, and information … will be updated each time the source file is modified; [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …)

15. Claims 19-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Murakami in view of Mandava et al., (Pat. No. US 7,203,928 B2) (hereinafter 'Mandava-2')

16. **As to claim 19** (original) (incorporating the rejection in claim 17), Murakami does not explicitly disclose the method wherein the file is an XML file.

However, in an analogous art of *Method and System for Generating and Maintaining Uniform Test Result*, Mandava-2 discloses the method wherein the file is an XML file (e.g., Fig. 2C; Fig. 2D; Fig. 3A; Col. 8, Lines 36-39 - … a companion static XML file …)

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-2 into the Murakami system to further provide the method wherein the file is an XML file in the Murakami system.

The motivation is that it would further enhance the Murakami's system by taking, advancing and/or incorporating Mandava-2's system which offers significant advantages to provide a flexible methodology and system for collecting and maintaining consistent test results generated as a result of a software application execution as once suggested by Mandava-2 (e.g., Col. 1, Lines 44-57)

17. **As to claim 20** (original) (incorporating the rejection in claim 19), Murakami discloses the method wherein the file comprises a pointer to at least one of a source

under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …), and Mandava-2 discloses requirement under test, and configuration under test (e.g., Col. 4, Lines 45-47 - … including a respective description entry explaining the function of each test case or test)

18. **As to claim 21** (original) (incorporating the rejection in claim 19), Mandava-2 discloses the method further comprising transforming the XML file utilizing XSLT (e.g., Fig. 2A, element 120 – XSLT Interface; Col. 8, Lines 23-25 - … using a Extensible Stylesheet Language (XSLT) Stylesheet interface) to enable management operations to be performed on the data including at least one of selection, query, reporting, suit composition, and scheduling (e.g., Fig. 1, elements 118 – Dynamic XML Results File, 120 – XSLT Interface)

19. Claims 23-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Mandava-2 in view of Murakami

20. **As to claim 23** (Currently Amended), Mandava-2 discloses a testing methodology comprising:

- loading a test case (e.g., Fig. 1, elements 104a - 104n; Col. 7, Lines 51-64) in accordance with a test ease file stored in a source file;

- executing the test case (e.g., Fig. 1, element 102 – target application; Col. 7, Lines 51-53 – a plurality of applications 104a – 104n for execution by a target application) on a source under test.

Mandava-2 discloses method for maintaining standardized test result (e.g., Abstract), but does not explicitly disclose the followings:

- generating test results, wherein the test results are version tagged to indicate the relationships between test results, version of the test case, and version of the source code under test; and
- continuously modifying test information such that new features are added and/or removed to test version changes to the source code under test.

However, in an analogous art of *Program Product, Method, and System for testing Consistency of Machine Code Files and Source Files*, Murakami discloses the followings:

- generating test results, wherein the test results are version tagged to indicate the relationships between test results, version of the test case, and version of the source code under test; and
- continuously modifying test information such that new features are added and/or removed to test version changes to the source code under test.

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Murakami into the Mandava-2's system to further provide the followings in the Mandava-2 system:

- generating test results, wherein the test results are version tagged to indicate the relationships between test results, version of the test case, and version of the source code under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …); and
- continuously modifying test information such that new features are added and/or removed to test version changes to the source code under test (e.g., [0008], Lines 6-9 – … to show the association between a new executive file version and its corresponding source file version …; [0049], Lines 5-9 - … inserts some additional information to the object file when building a library … for future consistency management; [0007], Lines 5-11 – those library resources, however, are modified for various reasons … implementation of additional capabilities … the modified source file is then recompiled, and the resulting new object file is used to create a new version of executive files).

The motivation is that it would further enhance the Mandava-2's system by taking, advancing and/or incorporating Murakami's system which offers significant advantages of an automated method that helps the users to test whether their local library resources reflect all modifications made to source files as once suggested by Murakami (e.g., Col. 1, Lines 1-6)

21. **As to claim 24** (original) (incorporating the rejection in claim 23), Mandava-2 discloses the method further comprising saving test results to an XML file (e.g., Fig. 1,

elements 114 – Dynamic XML File, 118 – Dynamic XML Results File; Fig. 2A, elements 114 – Dynamic XML File, 116 – Dynamic XML Results File; Fig. 4, elements 114"a – 114"c, 118" – Merged Dynamic XML Results File).

22. **As to claim 25** (original) (incorporating the rejection in claim 23), Mandava-2 discloses the method further comprising publishing the test results to an enterprise data store (e.g., Fig. 4, elements 124 – Report Tool, 126 – DB).

23. **As to claim 26** (original) (incorporating the rejection in claim 23), Mandava-2 discloses the method wherein the version tags indicate the version of the source under test and the version of the test (e.g., Col. 2, Lines 36-46 – matching each test case in the dynamic file with a corresponding test case in the static file…; Fig. 3E; Col. 18, Lines 43-53; Fig. 9; Col. 20, Lines 18-27).

24. **As to claim 27** (original) (incorporating the rejection in claim 23), Mandava-2 discloses a computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 23 (e.g., Col. 21, Line 45 through Col. 24, Line 13).

25. Claims 7-8, 10-12, and 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Murakami in view of Cowan and further in view of Mandava-2

26. **As to claim 7** (original) (incorporating the rejection in claim 1), Murakami and Cowan do not explicitly disclose the system wherein the test case file component is an XML document.

However, in an analogous art of *Method and System for Generating and Maintaining Uniform Test Result*, Mandava-2discloses the system wherein the test case file component is an XML document (e.g., Fig. 2C; Fig. 2D; Fig. 3A; Col. 8, Lines 36-39 - … a companion static XML file …)

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-2 into the Murakami-Cowan system to further provide the system wherein the test case file component is an XML document in the Murakami-Cowan system.

The motivation is that it would further enhance the Murakami-Cowan's system by taking, advancing and/or incorporating Mandava-2's system which offers significant advantages to provide a flexible methodology and system for collecting and maintaining consistent test results generated as a result of a software application execution as once suggested by Mandava-2 (e.g., Col. 1, Lines 44-57)

27. **As to claim 8** (Previously Presented) (incorporating the rejection in claim 7), Mandava-2 discloses the system wherein the XSLT is employed to facilitate management operations including at least one of selection, query, reporting, suit composition, and scheduling (e.g., Fig. 2A, element 120 – XSLT Interface; Col. 8, Lines 23-25 - … using a Extensible Stylesheet Language (XSLT) Stylesheet interface)

28. **As to claim 10** (Currently Amended) (incorporating the rejection in claim 9), Mandava-2 discloses the system wherein the test case file component is loaded into the test catalog (e.g., Fig. 1, element 110 – Static XML File)

29. **As to claim 11** (original) (incorporating the rejection in claim 8), Mandava-2 discloses the system wherein the test case component specified in the test case file component is loaded into the test catalog (e.g., Fig. 1, element 110 – Static XML File)

30. **As to claim 12** (original) (incorporating the rejection in claim 11), Mandava-2 discloses the system wherein a test execution component executes the test case on the software under test and generates test results (e.g., Fig. 1, element 102 – target application; Col. 7, Lines 51-53 – a plurality of applications 104a – 104n for execution by a target application; Fig. 1, element 114 – dynamic XML file)

31. **As to claim 15** (original) (incorporating the rejection in claim 14), Mandava-2 discloses the system wherein the means for maintaining fine-grained track of a test's relation to a version of software under test includes persisting software version information and related test information to an XML file (e.g., Fig. 2C; Fig. 2D; Fig. 3A; Col. 8, Lines 36-39 - … a companion static XML file …)

32. **As to claim 16** (original) (incorporating the rejection in claim 15), Mandava-2 discloses the system wherein the XML file is transformed utilizing XSLT to enable test

data to be queried (e.g., Fig. 2A, element 120 – XSLT Interface; Col. 8, Lines 23-25 - … using a Extensible Stylesheet Language (XSLT) Stylesheet interface)

Conclusion

33. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ben C. Wang whose telephone number is 571-270-1240. The examiner can normally be reached on Monday - Friday, 8:00 a.m. - 5:00 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Ben C Wang/
Examiner, Art Unit 2192

May 21, 2008

/Eric B. Kiss/

Primary Examiner, Art Unit 2192